

Министерство науки и высшего образования Российской Федерации  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Институт прикладной математики и компьютерных наук

УТВЕРЖДАЮ:

Директор



А. В. Замятин

« 15 » июня 20 23 г.

Рабочая программа дисциплины

**Разработка программного обеспечения и скриптовые языки**

по направлению подготовки

**01.04.02 Прикладная математика и информатика**

Направленность (профиль) подготовки :

**Интеллектуальный анализ больших данных**

Форма обучения

**Очная**

Квалификация

**Магистр**

Год приема

**2023**

Код дисциплины в учебном плане: Б1.О.03.02

СОГЛАСОВАНО:

Руководитель ОП

А.В. Замятин

Председатель УМК

С.П. Сущенко

Томск – 2023

## **1. Цель и планируемые результаты освоения дисциплины**

Целью освоения дисциплины является формирование следующих компетенций:

- УК-2 – способность управлять проектом на всех этапах его жизненного цикла;
- ОПК-4 – способность комбинировать и адаптировать существующие

информационно-коммуникационные технологии для решения задач в области профессиональной деятельности с учетом требований информационной безопасности.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИУК-2.3 Обеспечивает выполнение проекта в соответствии с установленными целями, сроками и затратами.

ИУК-2.2 Разрабатывает программу действий по решению задач проекта с учетом имеющихся ресурсов и ограничений.

ИУК-2.1 Формулирует цель проекта, обосновывает его значимость и реализуемость.

ИОПК-4.3 Использует современные информационно-коммуникационные технологии для решения задач в области прикладной математики и информатики с учетом требований информационной безопасности.

ИОПК-4.1 Анализирует задачи прикладной математики и информатики средствами информационных технологий.

## **2. Задачи освоения дисциплины**

– Формирование навыков для профессионального программирования с использованием современных языков и их реализаций, встроенных и сторонних библиотек и пакетов программ; ознакомление с современными профессиональными стандартами информационных технологий, получение детальных знаний парадигм и методологий программирования, особенностей языков программирования общего и специального назначения, таких как Python, Javascript, Perl; работа с разными инструментальными системами разработки программ.

## **3. Место дисциплины в структуре образовательной программы**

Дисциплина относится к обязательной части образовательной программы. Дисциплина входит в модуль «Разработка программного обеспечения».

## **4. Семестр(ы) освоения и форма(ы) промежуточной аттестации по дисциплине**

Второй семестр, экзамен

## **5. Входные требования для освоения дисциплины**

Для успешного освоения дисциплины требуются результаты обучения по следующим дисциплинам: «Алгоритмы и структуры данных».

## **6. Язык реализации**

Русский

## **7. Объем дисциплины**

Общая трудоемкость дисциплины составляет 4 з.е., 144 часов, из которых:

-лекции: 16 ч.

-лабораторные: 16 ч.

в том числе практическая подготовка: 0 ч.

Объем самостоятельной работы студента определен учебным планом.

## **8. Содержание дисциплины, структурированное по темам**

### **Раздел 1. Общее введение в теорию компиляции**

- 1.1. Общее введение в теорию компиляции. Задачи и подходы трансляции.
- 1.2. Ликбез по компиляторам и принципам трансляции. Фазы компиляции

### **Раздел 2. Современные языки программирования**

- 2.1. Парадигмы и классификация языков программирования
- 2.2. Обзор современных языков программирования

### **Раздел 3. Разработка ПО с использованием скриптовых языков**

- 3.1. Основы методологии разработки ПО и организации командной работы
- 3.2. Основы разработки ПО на ЯП Python
- 3.3 Основы разработки ПО на ЯП Perl
- 3.4. Основы разработки ПО на ЯП Javascript

## **9. Текущий контроль по дисциплине**

Текущий контроль по дисциплине проводится путем проверки лабораторных работ.

- ЛЗ 1 . Практикум по языку Python
- ЛЗ 2 . Практикум по языку Perl.
- ЛЗ 3 . Практикум по языку JavaScript.

Примерный перечень вопросов текущего контроля:

### **Тема 1. Скриптовый язык Javascript**

1. Сколько параметров можно передать функции?
2. Чем отличается наследование в JavaScript от наследования в Python?
3. Что представляет из себя метод объекта в JavaScript?
4. Зачем в JavaScript перед переменной писать var?
5. Как создать массив в JavaScript?
6. Можно ли в JavaScript использовать функцию в качестве конструктора?
7. Какой тип у объекта NaN и что вернет выражение NaN=== NaN?
8. Какая разница между операторами == и ===?
9. С помощью какой конструкции языка можно управлять потоком выполнения скрипта и отслеживать ошибки?

### **Тема 2. Скриптовый язык Perl**

10. Какие средства управления исключительными ситуациями есть в Perl?
11. В чем особенность режима strict?
12. Как реализована поддержка принципов ООП?
13. Какие механизмы наследования есть в Perl?
14. В чем различие между функциями и методами?
15. Существует ли неявное преобразование числовых типов?
16. Является ли обращение к словарям атомарной операцией?
17. Операторы ++ - это выражения(expression) или конструкции(statement)?

### **Тема 3. Основы методологии разработки ПО и организации командной работы**

18. Какие классические методологии разработки ПО существуют?
19. Что такое гибкая (Agile) методология и какие ее разновидности приняты?
20. В чем отличие Роли от Активности?
21. Что такое Спринт и Журнал пожеланий спринта

## 10. Порядок проведения и критерии оценивания промежуточной аттестации

Промежуточная аттестация проводится в виде экзамена.

Типовые задания для проведения промежуточной аттестации по дисциплине

### Тема 1. Скриптовый язык Javascript

22. Сколько параметров можно передать функции?
23. Чем отличается наследование в JavaScript от наследования в Python?
24. Что представляет из себя метод объекта в JavaScript?
25. Зачем в JavaScript перед переменной писать var?
26. Как создать массив в JavaScript?
27. Можно ли в JavaScript использовать функцию в качестве конструктора?
28. Какой тип у объекта NaN и что вернет выражение NaN=== NaN?
29. Какая разница между операторами == и ===?
30. С помощью какой конструкции языка можно управлять потоком выполнения скрипта и отслеживать ошибки?

### Тема 2. Скриптовый язык Perl

31. Какие средства управления исключительными ситуациями есть в Perl?
32. В чем особенность режима strict?
33. Как реализована поддержка принципов ООП?
34. Какие механизмы наследования есть в Perl?
35. В чем различие между функциями и методами?
36. Существует ли неявное преобразование числовых типов?
37. Является ли обращение к словарям атомарной операцией?
38. Операторы ++ - это выражения(expression) или конструкции(statement)?

### Тема 3. Основы методологии разработки ПО и организации командной работы

39. Какие классические методологии разработки ПО существуют?
40. Что такое гибкая (Agile) методология и какие ее разновидности приняты?
41. В чем отличие Роли от Активности?
42. Что такое Спринт и Журнал пожеланий спринта

## 11. Учебно-методическое обеспечение

- а) Электронный учебный курс по дисциплине в электронном университете «Moodle»
- б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине.

## 12. Перечень учебной литературы и ресурсов сети Интернет

- а) основная литература:
  - Мозговой М.В. Классика программирования. Алгоритмы, языки, автоматы, компиляторы. Практический подход. – СПб.: Наука и Техника, 2006. – 320 с.
  - Вабищевич П.Н. Численные методы: вычислительный практикум: [практическое применение численных методов при использовании алгоритмического языка Python] . – М: Ленанд, 2015. – 319 с.: рис.

- б) дополнительная литература:
  - Fernández M. Programming Languages and Operational Semantics.[Электронный ресурс]. – London:Springer, 2014. - 209 с. 10 illus.: online resource. – URL: <http://dx.doi.org/10.1007/978-1-4471-6368-8> (дата обращения: 01.11.2021).

- в) ресурсы сети Интернет:

- Programming Languages - commonly used features in a side-by-side format [Электронный ресурс] / Hyperpolyglot.– 2016. – URL: <http://hyperpolyglot.org/> (дата обращения: 06.10.2016).
- The Computer Language Benchmarks Game [Электронный ресурс] / Alioth Debian .– 2016. – URL: <http://benchmarksgame.alioth.debian.org/> (дата обращения: 14.10.2016).
- The Scrum Guide [Электронный ресурс] / Ken Schwaber .– 2017. – URL: <https://www.scrum.org/resources/scrum-guide>
- Электронная библиотека (репозиторий) ТГУ [Электронный ресурс] . – Электрон. дан. – Томск, 2016- . URL: <http://vital.lib.tsu.ru/vital/access/manager/Index>

### **13. Перечень информационных технологий**

- а) лицензионное и свободно распространяемое программное обеспечение:
  - Онлайн интерпретатор языков Python3 и Javascript – Repl.it, онлайн интерпретатор языка Javascript – Tryit. Для языка Perl используется открытая и бесплатно распространяемая реализация strawberry perl 5.24.

### **14. Материально-техническое обеспечение**

Для реализации дисциплины необходимы лекционные аудитории и аудитории для проведения лабораторных занятий. Специальные технические средства (проектор, компьютер и т.д.) требуются для демонстрации материала в рамках изучаемых разделов, проведения защиты проектов в конце семестра. Вся основная и дополнительная литература, необходимая для самостоятельной работы и подготовки к экзамену, имеется в научной библиотеке ТГУ.

### **15. Информация о разработчиках**

Овсянников Михаил Сергеевич, старший преподаватель кафедры теоретических основ информатики ТГУ.